GreenSlice: An Energy-Efficient Secure Network Slicing Framework

Ozan Akin*, Umut Can Gulmez*†, Ozan Sazak*, Osman Ufuk Yagmur*, and Pelin Angin
Middle East Technical University, Ankara, Turkey
{ozan.akin, umut.gulmez, ozan.sazak, ufuk.yagmur}@metu.edu.tr, pangin@ceng.metu.edu.tr

Received: July 31, 2021; Accepted: December 29, 2021; Published: February 28, 2022

Abstract

The fifth generation of telecommunication networks comes with various use cases such as Enhanced Mobile Broadband, Ultra-Reliable and Low Latency Communications and Massive Machine Type Communications. These different types of communications have diverse requirements that need to be satisfied while they utilize the same physical infrastructure. By leveraging Software Defined Network (SDN) and Virtual Network Function (VNF) technologies, the 5G network slicing concept can provide end-to-end logical networks on the same physical infrastructure that satisfy the required Quality of Service (QoS) constraints for these communication types. Optimal placement of VNFs on these network slices is still an open problem. Although state-of-the-art research covers the resource allocation of these VNFs, they do not consider optimizing energy consumption under strict security requirements while embedding them into the network. In this paper, we propose a VNF placement strategy using an integer linear programming (ILP) model for 5G network slicing under strict security requirements, which optimizes energy consumption by the core network nodes. Simulation results demonstrate that the proposed model achieves significant power savings over a greedy approach performing VNF placement under the same QoS and security constraints.

Keywords: Network Slice, ILP, Slice Embedding, Security, 5G

1 Introduction

The Internet of Things (IoT), enabling the connectivity of physical and virtual objects to create smart environments, has witnessed exponential growth in the past decade with the advances in networking infrastructures and smart devices. Among major applications of IoT are *smart homes* integrating various sensors for security, elderly care and smart energy consumption, *wearables* for personal health monitoring, *smart manufacturing* expected to be prevalent in the logistic chain and production line, *smart energy grids*, *smart cities* utilizing data from various sensors for long term development planning, *connected vehicles*, *smart farming* improving the agri-food supply chain, *earth/ocean observation systems* addressing environmental issues, and *surveillance/safety warning systems* for emergency response. The wide adoption of IoT is expected to provide immense economic benefits in the whole world through the creation of a sustainable ecosystem, as it enables crossing over borders between different industrial sectors, creating more efficient processes, reduced consumption, and increased sharing of resources, despite bringing along an enlarged attack surface [1].

Journal of Internet Services and Information Security (JISIS), volume: 12, number: 1 (February), pp. 57-71
DOI:*10.22667/JISIS.2022.02.28.057*

*These authors contributed equally to this work

†Corresponding author: Wireless Systems, Networks, and Cybersecurity Lab, Middle East Technical University, Computer Engineering Department, B Building, Z-019, Cankaya/Ankara, Turkey, Tel: +90-539-916-2746

As a major difference from existing mobile networks, 5G networks that will support various IoT use cases and mobile cloud computing [2] will significantly rely on **network virtualization** technologies including *software-defined networking* (SDN) and *network functions virtualization* (NFV) for effective dynamic resource allocation and end-to-end (E2E) operation. Accordingly, network slicing has arisen as a concept to provide the required flexibility for the operators in managing their resources effectively to provide the desired quality of service (QoS) guarantees to their customers. Through creating multiple virtual networks called slices on top of shared physical infrastructure, network slicing will enable the creation of isolated networks, allowing the operators to dynamically provision only resources necessary to support the requirements of specific IoT applications, providing effective utilization of the physical infrastructure, hence cost savings. Network slicing is expected to enable enhanced security management through **isolating traffic** for different applications, preventing attacks such as denial of service (DoS) on a particular slice from affecting the traffic on other slices.

While network slicing is seen as a compulsory technology for the successful operation of next-generation mobile networks, its effective implementation faces many challenges to be solved before widespread adoption by telecommunication operators can take place:

- **Security** is a critical problem [3], [4] due to sharing of physical resources between slices, each of which may have different security requirements [5]. The security requirements of one slice could affect the overall performance of other network slices.
- Given the concerns regarding the huge energy consumption expectations for 5G networks, one of the key performance indicators will be **energy efficiency** [6] as in other wireless networks [7]. Jointly optimizing energy consumption and QoS parameters like delay, bandwidth and throughput while meeting **strict security requirements** is a challenge, as these are conflicting in most cases.
- Dynamically embedding slices onto a network topology is challenging; operators need to consider changing requirements of the network slices themselves, e.g., traffic volume and QoS requirement may change [8].

To address the above challenges, this paper proposes a novel energy-efficient and secure network slice embedding model for 5G core networks with Integer Linear Programming (ILP) based optimization.

The major contributions of this paper to the literature on secure network slicing are as follows:

- We propose an optimization model for the virtual network embedding problem in 5G core network slicing that achieves significant power savings in overall core network power consumption under strict security requirements for virtual network function placement.
- We provide a network topology generator that can be used to test optimization models on a variety of topologies with different requirements.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in energy-efficient and secure network slicing. Section 3 introduces the concept of network slicing and its enablers, including NFV and SDN. Section 4 describes our proposed approach for energy-efficient, secure core network slicing. Section 5 provides an evaluation of the proposed approach with realistic simulation experiments. Section 6 concludes the paper with future work directions.

2 Related Work

Network slicing is a new technology that has been proposed to support effective management of 5G networks and beyond, for which many research and development efforts are still in progress, with no mature standards in place. 5G is expected to provide increased performance as compared to the previous generation of mobile networking technologies. As the utilized bandwidth gets wider, connected devices get larger in number, and data rates become faster, the energy needed for these operations will also increase. Therefore, it is an important research problem to optimize the energy consumption of 5G networks while providing the required services with the required QoS. Below we provide an overview of existing approaches that address the aspects of network slicing that we focus on in this work.

In [9] a security-aware slice instance allocation model for 5G core networks was proposed. Security limitations such as some of the VNFs having to be hosted on the same server and some of the VNFs not being able to coexist on the same server were given to the ILP solver as constraints. Their work showed that there is a trade-off between slice security and embedding performance metrics such as execution time and average revenue cost ratio for accepted requests. Although they considered security aspects of resource allocation, they did not look at energy efficiency. [10] handles virtual network security functions (VNSFs) placement as an ILP problem while considering security and QoS requirements of the network slices. For this purpose, they give total maximum end-to-end latency as a QoS constraint and VNSF execution order, VNSF network position, and operational mode as security constraints. However, energy efficiency optimization was not considered in the model. Guan et al. [11] also implemented an algorithm that places VNSFs onto a network topology using routing characteristics instead of ILP, and tested their security performance in a simulation that mimics computer virus and worm attacks on the network.

There has been research being conducted on energy efficiency in 5G networks in every layer of the 5G architecture, from base stations to radio access network (RAN) and to core 5G networks. In this research, our focus is on achieving an energy-efficient secure network slicing scheme on the core network. Energy efficiency in networking is usually considered as a fractional programming problem since providing more service with as little energy as possible is, by its very nature, a trade-off problem. Therefore, researchers have approached the energy efficiency issue as a ”fraction” to be maximized, where the numerator and denominator are two sides of a trade-off. Nguyen proposed in [12] a hybrid resource allocation scheme that considers spectrum allocation, interference alignment, and energy efficiency simultaneously since all three of them are important for providing good performance in the network. In [13], Matthiesen et al. developed a QoS framework for a sliced radio network (RAN), where two network parameters were considered: throughput and energy efficiency. They built Pareto boundaries of two different algorithms, which are based on utility profile and scalarization, respectively. In [14], the researchers focused on the energy efficiency vs. delay trade-off problem in wireless network virtualization. They modeled the issue as a stochastic optimization problem with predefined delay constraints, where users are queued on virtual base stations.

Mathematical optimization is not the only approach used in optimizing energy consumption in networking. With the advancements in hardware technology and data science in recent years, reinforcement learning-based models have proven successful in network resource allocation and optimization. [15] proposed an algorithm that considers both energy efficiency and spectral efficiency of the network while using Dueling Deep Q-Network and shows successful results compared to Q-learning and DQN. In [16], base stations' sleep modes were optimized with the help of Q-Learning, a variant of reinforcement learning. The aim was to find the optimal operating duration of base stations with respect to the delay and energy consumption requirements and activate and deactivate them accordingly. Laroi et al. [17] developed a VNF slice placement algorithm for core networks using Deep Reinforcement Learning (DRL) algorithms as well as an ILP algorithm and compared their performances. Results showed that the DRL model consumes less energy and time than ILP and reinforcement learning algorithms. However, they did not consider the security aspects of VNF placement. Particularly in network slicing, there have been many works optimizing the resource allocation between network slices [18], [19], [20], though not particularly on optimizing the energy consumption in network slices. Fendt et al. [8] proposed a model that handles slice instance mapping onto a given network topology using ILP. They focused on network slice instance and link embedding, and resource allocation to these slices. In their work, link capacity, latency, and graph constraints were given as constraints to the ILP model.

Although there exist approaches focusing on optimal network slicing/virtual network embedding, none of them consider both energy efficiency and security constraints of the network. As we show later in the paper, one should consider security while optimizing network slicing since security comes with a cost. Our approach handles the VNF allocation problem more comprehensively than other approaches in the state-of-the-art. It provides a solution for the secure slice embedding problem while minimizing energy consumption.

3 Preliminaries

Based on the definition by the 3GPP foundation, **a network slice** is a logical end-to-end network that can be created on demand on a physical network infrastructure, where users can access multiple slices over the same radio interface [21]. These slices contain **network slice instances** that consist of several network functions. **Network functions** have a detailed functional behavior and well-defined 3GPP interfaces [21]. These logical networks will provide various use cases like enhanced mobile broadband, ultra reliable low latency communications and massive machine type communications. Figure 1 shows the network slicing concept with several use cases of the slices and network slice instances.

SDN and NFV are the key technologies for network slicing [22]. **NFV** is the virtualization of network functions on the shared hardware, such as firewalls, VPNs and 5G Evolved Packet Core functions. ETSI has published a standard [23] for NFV, which is widely accepted in both academia and industry. Figure 2 demonstrates the three layers of network slices. In the lower layer, which is called VNFI, there is the physical infrastructure of the network. This layer consist of storage, computing and network hardware, hypervisors and virtual machines that run on the physical infrastructure. The middle layer is virtualized network functions (VNFs)[21]. The uppermost layer is called Operational Support Systems and Business Support System (OSS/BSS). The OSS/BSS layer consists of management functions for network operators such as inventory, service provisioning, network configuration and fault management. These layers are managed by the Management and Orchestration (MANO) component of the network.

Software defined networks (SDN) decouple the data plane from the control plane, as opposed to legacy networks, which have a strong coupling between these two layers. While the control plane makes decisions regarding the routing of packets over the network, the data plane is only responsible for forwarding packets based on the flow rules decided by the control plane. One of the essential benefits of using SDN is that it significantly decreases the capital expenses of the operator. Since the control plane is separated and usually runs in the cloud, operators can use less expensive switches in the transport layer. SDN also provides advantages for the security management of the network. As the controller can monitor the whole traffic in the network, it can more easily detect anomalies and hence maintain the security of the network. In addition to that, due to the centralized control in SDN, any security policy configuration could be deployed more efficiently compared to legacy decentralized networks [22].

4 Proposed Network Slicing Approach

This section introduces the proposed ILP model for secure VNF placement in core network slicing, which aims to minimize power consumption while meeting the given memory, throughput and latency requirements. Our model is an extension of the model proposed by Fendt et al. [8]. The primary purpose of the VNF placement problem is mapping VNFs into servers in the network in the most effective manner. One of the main challenges in this virtual network embedding problem is solving this embedding

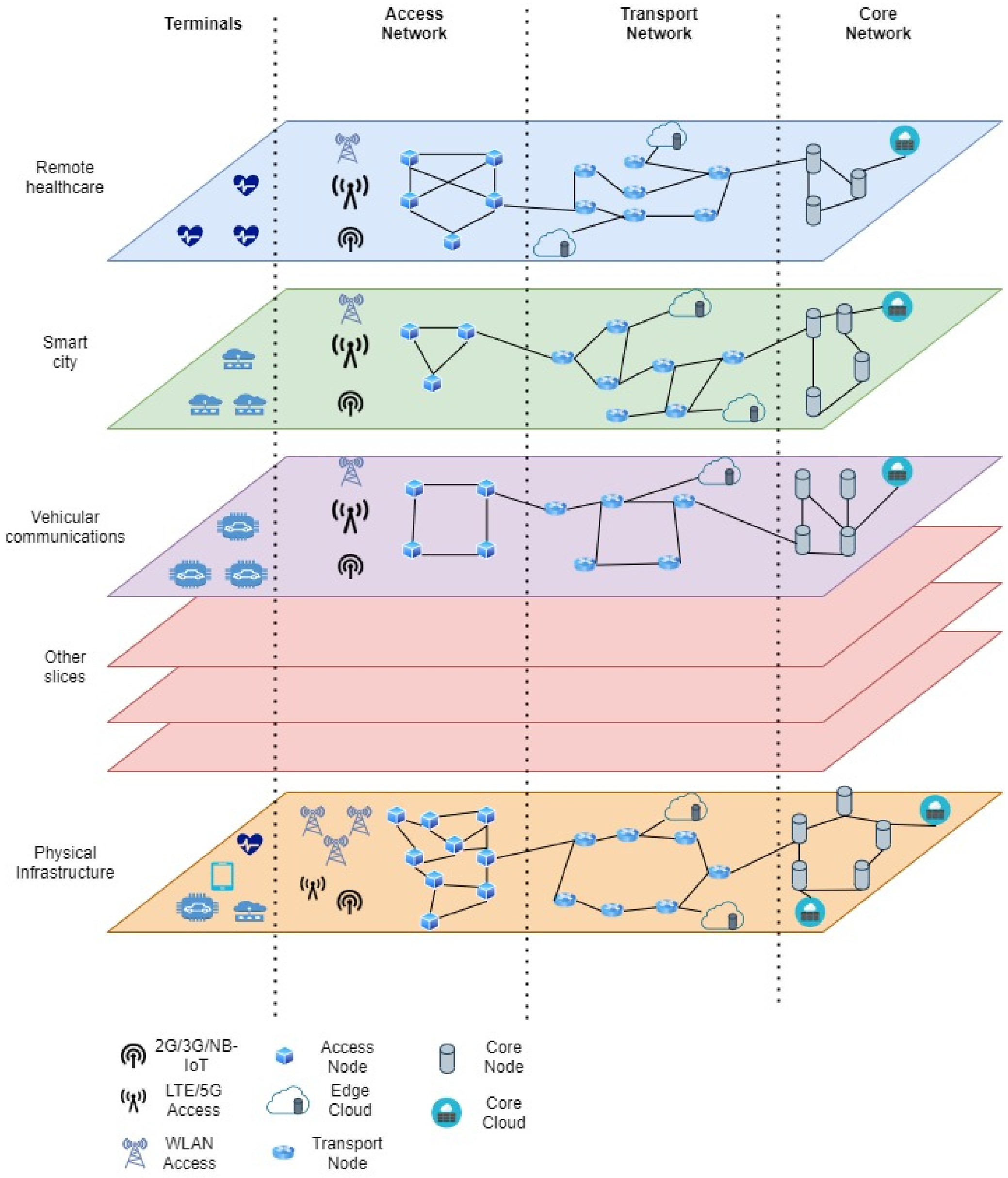


Figure 1: Network Slicing Architecture

problem in a security-aware way. The presented mathematical model can be used with an Integer-Linear Programming Solver, such as Gurobi[1] which is used in this paper.

4.1 Definitions

In this section, important parameters and definitions are covered, before delving into the details of the ILP model.

An undirected graph G is an ordered pair (V,E), in which V stands for the set of v nodes in the graph, and E stands for the edges in the graph. Each e_{ij} represents an edge between two nodes v_i and v_j that are

[1] Gurobi - The Fastest ILP Solver https://www.gurobi.com/

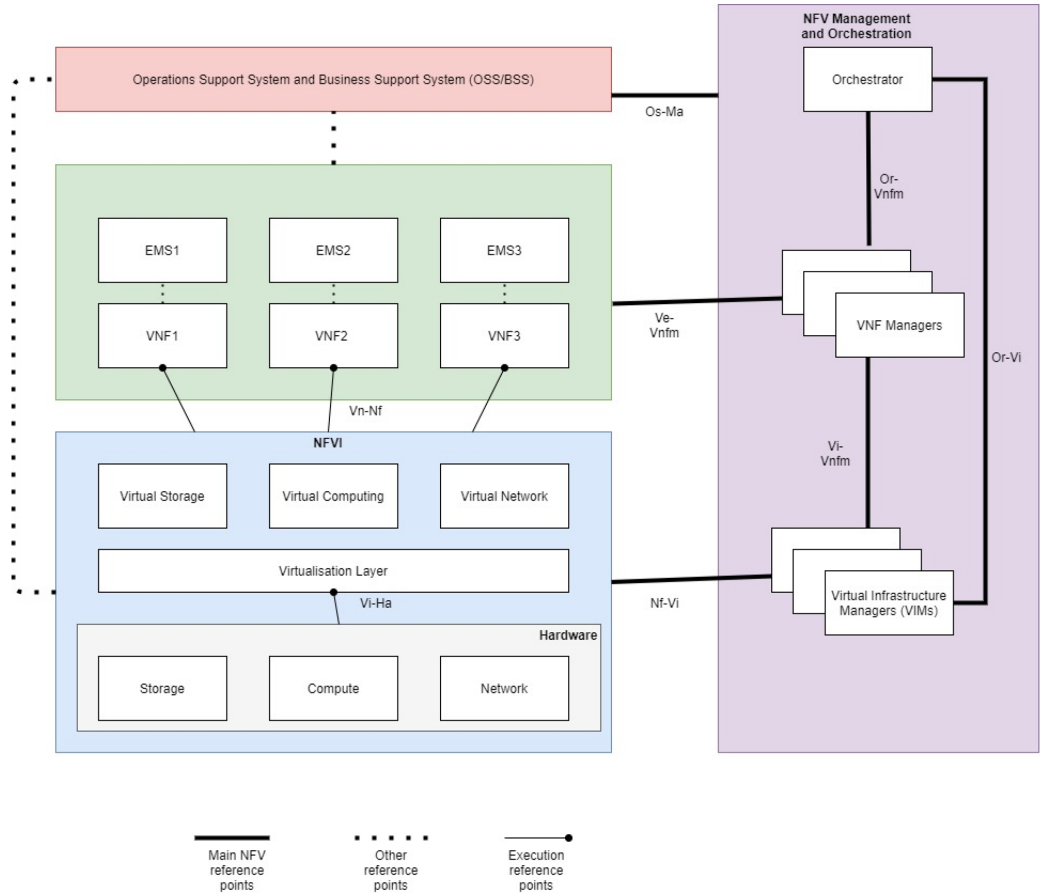


Figure 2: ETSI NFV architecture

connected with each other.

$$V = \{v_1, v_2, ..., v_k\} \tag{1}$$

$$E = \{e_{12}, e_{23}, ..., e_{ij}\} \text{ , where } v_i, v_j \in V \tag{2}$$

$$P_{ij} = \{e_{ir}, ..., e_{tj}\}, \text{ where } v_i, v_r, v_t, v_j \in V \tag{3}$$

P, an ordered set of nodes, describes a path in the graph G. Note that P is a subset of V. P_{ij} defines a path between nodes v_i and v_j.

An illustration of an example network graph can be found in Figure 3, and Figure 4 shows an example placement of a VNF in the network. The colored lines between user equipment nodes and VNF shows the paths used for connections between them. The paths are organized using the constraints and requirements, and this is why the paths used are not the shortest paths. For example, although there is a shorter path `U-2, S-8, VNF` in Figure 4 between node `U-2` and `VNF`, our algorithm picked the path `U-2, S-8, S-1, VNF` to meet the constraints and requirements.

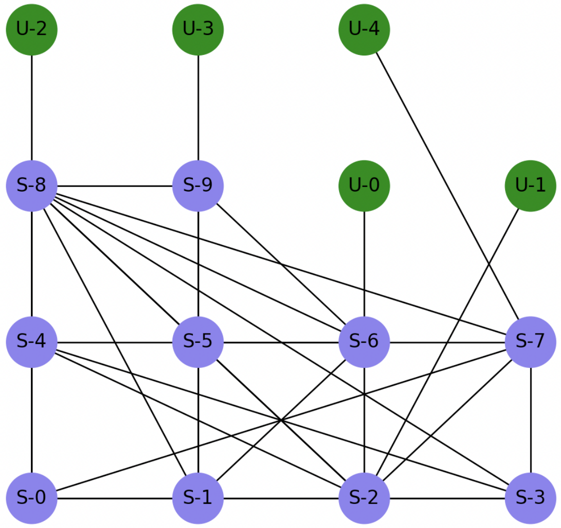


Figure 3: Example network that consists of server and user equipment nodes

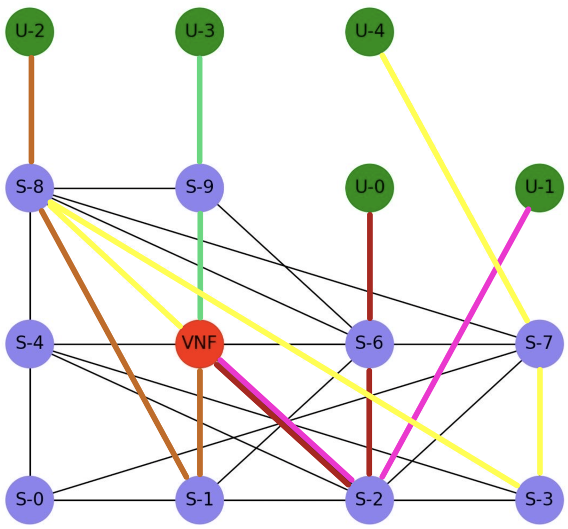


Figure 4: Example VNF placement that consists of paths between VNF and user equipment nodes

4.2 Model Parameters

In the proposed model, it is assumed that a physical network contains a number of server nodes and user equipment nodes. The main target of this solution is embedding VNFs into the server nodes in a way that optimizes the total power consumption under the given security requirements and QoS constraints.

The set Δ defines the list of VNFs which will be used in the network slices.

$$\Delta = \{\delta_0, \delta_1, ..., \delta_m\} \quad (4) \qquad L_k = \{l_0, l_1, ..., l_i\} \quad (5) \qquad N_k = (V, E, \Delta, L) \quad (6)$$

Let's call a VNF δ, and let Δ be a set of VNFs. We can define the kth network slice, namely $N_k = (V, E, \Delta, L)$, where V stands for nodes, E for physical links, and Δ for virtual network functions, and L for virtual links that connect user equipment nodes with server nodes. A virtual link $l \in L$ is a set of edges (e) that connects a user equipment node to a VNF running in a server node in the graph.

$$F_k = \{\delta_r, ..., \delta_m\} \quad (7) \qquad LS_k = \{\delta_r, ..., \delta_m\} \quad (8)$$

The equations 7, 8 are used for providing security constraints, which will be covered in the following section.

The parameters of the model are given below:

- N_k: kth network slice.
- s_w: wth server node.
- u_v: vth user equipment node.
- e_j: jth physical link.
- F_o: Forbidden set of VNF o.

- LS_m: Locate-Same set of VNF m.

The parameters below are the limits which will be used during the optimization process.

- M^s_w: maximum memory of the server node s_w.
- R^s_w: maximum number of requests of the server node s_w.
- P^s_w: power consumption of a virtual network function running in the server node s_w.
- T^e_j: maximum throughput of the physical link e_j.
- L^e_j: maximum latency of the physical link e_j.
- $\delta_{k,m}$: mth VNF which runs in the kth slice.
- $l_{k,i}$: ith virtual link in kth slice, which connects a user equipment node and a VNF. This value is computed using the the sets s_w, u_v, and $\delta_{k,m}$.
- $P^{\delta}_{k,m}$: the additional power consumption of the VNF δ_m in the kth network slice, in addition to P^s_w.
- $M^{\delta}_{k,m}$: the memory usage of the VNF m in the kth network slice.
- $R^{\delta}_{k,m}$: the average requests usage of the VNF m in the kth network slice.
- $T^l_{k,i}$: the instantaneous throughput of the ith virtual link in kth slice, $l_{k,i}$.
- $L^l_{k,i}$: the maximum latency of the ith virtual link in the kth slice, $l_{k,i}$.

These are the static parameters that are a part of the graph and constraints and cannot be changed after the infrastructure has been set. In addition, there are some related dynamic parameters that changes in the embedding process.

The variable $\Phi_{r,j}$ defines a physical link e_j used for constructing P_r

$$\Phi_{r,j} = \begin{cases} 1 & \text{if } P_r \text{ uses } e_j \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

The integer variable in the equation 9 is not a part of the proposed ILP model. It has no effect on the proposed solution. On the other hand, this variable is changed after the optimized solution is found by the model. It defines whether path P uses physical edge e_j. By using this variable, the visualized version of the network topology is created, which will be demonstrated below.

4.3 Decision Variables

In the proposed ILP model, there exist binary and linear decision variables whose values change during the optimization process, which are the key points of ILP. By changing the values of decision variables, the ILP model tries to find the optimized solution from among the feasible solutions.

Here are some variables that have been adapted from [8] to our model:

$$\mu_{k,m,w} = \begin{cases} 1 & \text{if } \delta_{k,m} \text{ is mapped on } s_w \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$\rho_{k,i,r} = \begin{cases} 1 & \text{if } P_r \text{ is used in } l_{k,i} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

The integer variable in Equation 10 decides whether a virtual network function δ is mapped on a server node s_w. In addition, the variable in Equation 11 does the same operations on virtual links over a physical path.

4.4 Objective Function

As mentioned earlier, the purpose behind the objective function is minimizing the power consumption while mapping VNFs and server nodes by providing the requirements and limitations. In Equation 12, there exists a static power consumption of every VNF if it is mapped to a server node, which increases the overall power consumption of the network.

$$min \sum_k \sum_m \sum_w [(P^s_w + P^{\delta}_{k,m}) \cdot \mu_{k,m,w}] \tag{12}$$

4.5 Optimization Constraints

4.5.1 Graph constraints

$$\sum_w \mu_{k,m,w} = 1 \ \forall k,m \tag{13}$$

$$\sum_{P_r} \rho_{k,i,r} = \mu_{k,m,w} \ \forall k,i \ \text{ where } \ l_{k,i} \text{ is a link from } s_w \text{ to } \delta_m \tag{14}$$

The constraint in Equation 13 maps every virtual network function to a server node in the network slices so that every virtual network function is mapped to that slice at least once.

The constraint in Equation 14 ensures that the virtual links that connect virtual network functions in slices and user equipment nodes are synchronized with the paths.

4.5.2 Capacity constraints

$$\sum_k \sum_i [(\sum_r \rho_{k,i,r} \cdot \Phi_{r,j}) \cdot T^l_{k,i}] \leq T^e_j, \forall j \tag{15}$$

$$\sum_k \sum_m (\mu_{k,m,w} \cdot M^{\delta}_{k,m}) \leq M^s_w, \forall w \tag{16}$$

$$\sum_k \sum_m (\mu_{k,m,w} \cdot R^{\delta}_{k,m}) \leq R^s_w, \forall w \tag{17}$$

The constraints above guarantee that if a VNF is mapped on a server node, that server node must meet the QoS needs of the VNF. In addition, these requirements cannot exceed the limitations of server nodes. Every server node has predefined throughput, power usage, and memory usage limits. These are given as static data. Every VNF has a throughput requirement that needs to be provided by server nodes. The constraint in Equation 15 ensures that the throughput on every virtual link does not exceed its maximum value. In addition to that, for every VNF, there exist memory requirements that should be maintained by the mapped server node. This gives the constraint in Equation 16, which ensures the maximum memory usage does not exceed its maximum possible value. Similarly, Equation 17 ensures for each server node, the maximum required number of requests does not exceed the total number of requests in the server node.

4.5.3 Latency constraints

$$\sum_{j}[\sum_{r}(\rho_{k,i,r} \cdot \Phi_{r,j}) \cdot L^{l}_{j}] \leq L^{e}_{k,i} \ ,\forall k,i \tag{18}$$

One of the critical points in network slicing is that while mapping the server nodes and network functions, the latency requirements must be met. The constraint in Equation 18 ensures that the latency in every virtual link is in the required range.

4.5.4 Security Constraints

$$\mu_{k,m,w} \cdot \mu_{k,o,w} = 0 \ , \ \forall o \in F_m \tag{19}$$

$$\mu_{k,m,w} = \mu_{k,o,w} \ , \ \forall o \in LS_m \tag{20}$$

Inspired by [9], to meet security requirements, two different static sets are defined, including Forbidden and Locate-Same. If VNF *m* contains VNF *o* in its forbidden set, then these VNFs will not be placed on the same server node. If VNFs create a security vulnerability for each other or for a server node, then the Forbidden sets (Equation 19) of those VNFs will contain other VNFs that create a vulnerability. The Locate-Same set works entirely in the opposite way. VNFs may need other network functions to have a secure environment. For example, VNFs may require a firewall in the same server node to have a safe environment. Then, in the Locate-Same sets (Equation 20), these VNFs will contain the network functions that are required for them. These are static sets that never change while the proposed solution is running. These sets will change the placements of VNFs so that they will meet the security requirements given with these Forbidden and Locate-Same sets.

4.6 ILP Model

The proposed model uses the constraints given in the previous section and creates the network topology by optimizing the objective function. While doing this, graph, capacity, latency, and security constraints are maintained by the optimizer. The network topology generated using the ILP model is a system that meets all the requirements and constraints and also a system that has minimum power consumption in given parameters. In addition to that, the wanted VNF types, the memory, power, or latency constraints can be defined for each server node, physical link, or slice separately.

5 Experimental Evaluation

To evaluate the ILP algorithm we introduced in the previous section, we have developed a simulation framework using the Gurobi integer linear programming simulation library. We have used Python for implementing the algorithm and the simulation environment. We have run the benchmarks on a MacBook Pro 16, 2.6 GHz 6-Core Intel Core i7 with 32 GB RAM. We have also developed a dataset generator to run the simulations on our own generated comprehensive, fully customizable datasets. Below we present the details of the evaluation environment and the results.

Table 1: Dataset generation parameters used for the simulations

Count of slices	2	Min memory per VNF	16
Count of edge nodes	2	Max memory per VNF	2048
Min number of VNFs per slice	10	Min request count per VNF	100
Max number of VNFs per slice	20	Max request count per VNF	1000
Min additional power usage per VNF	10	Min power usage per server node per VNF	1000
Max additional power usage per VNF	500	Max power usage per server node per VNF	50000
Min latency per physical link	5	Min memory count per server node	2048
Max latency per physical link	25	Max memory count per server node	10240
Min throughput per virtual link	100	Min request count per server node	10000
Max throughput per virtual link	500	Max request count per server node	100000
Min latency per virtual link	50	Min throughput per physical link	10000
Max latency per virtual link	150	Max throughput per physical link	20000

5.1 Evaluation Environment

In the experimental evaluation, we used various simulation settings with different parameters. The parameters used in the simulation are listed in Table 1. Each configuration was used five times to generate 50 different datasets in total, and we took the average of the metric values we obtained from the simulations.

As a baseline for comparison, we also implemented a greedy approach in Python that performs VNF placement in the network topology. The greedy approach places VNFs in server nodes with the given capacity, security, and graph constraints, without optimizing power consumption. Moreover, to measure the cost of the power optimization during the ILP-based optimization process, we have also run our ILP model on 10 of our datasets without an objective function, which basically does VNF placement under the given constraints.

To verify the correctness of our ILP model and greedy approaches, we have also built a verifier that checks each graph, security, and capacity constraint. The verification code confirms that both the ILP and the greedy algorithms are compatible.

5.2 Evaluation Results

We have compared the evaluation results of the greedy and ILP algorithms, and the comparison of these two algorithms both from the aspect of time and power can be seen in Figure-5 and Figure-6. The line plot in Figure-5 shows the change of problem-solving time with respect to the server node count. As it can be seen from Figure-5, the build time of the ILP algorithm is much more than the optimization time. In addition, when the server node count increases, the build time of this model is also increased. On the other hand, when the server node count increases, the time difference between ILP optimization and the greedy algorithm decreases. The bar plot in Figure-6 shows the total power consumption difference between the ILP and greedy algorithms vs. server node count. The greater the value, the better the ILP algorithm is in outperforming the greedy algorithm.

As can be seen in the optimization time versus server count on the plot in Figure-5, both of the algorithms seem to follow an exponential time complexity for finding the solutions. Also, there is a constant level of difference between the data points of both algorithms, and we can say this difference is because of the model building time of Gurobi, the ILP framework that we have used.

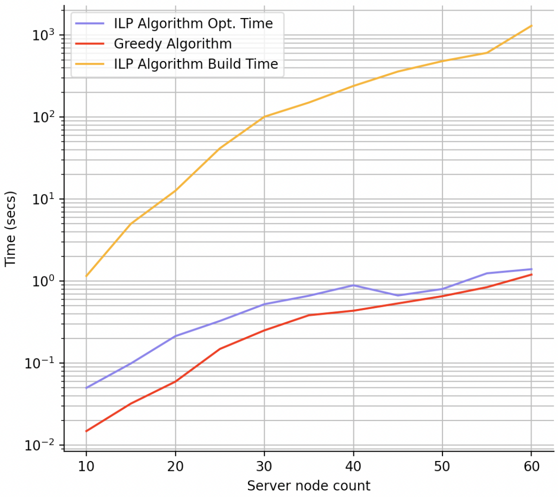


Figure 5: Total problem solving time by server node count plot, in logarithmic scale

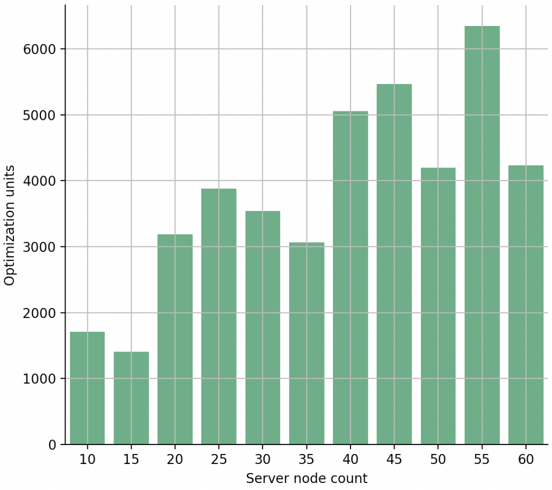


Figure 6: Optimization result difference between Greedy Implementation and ILP Implementation by server node count plot

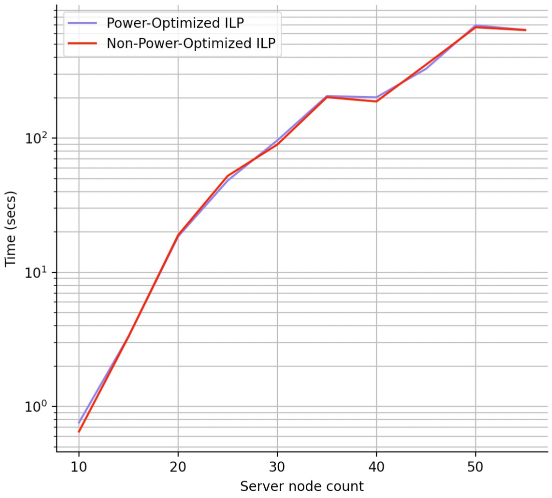


Figure 7: Total problem solving times (model build time + model optimization time) for non-power optimized and power-optimized ILP implementation, by server node count plot, in logarithmic scale

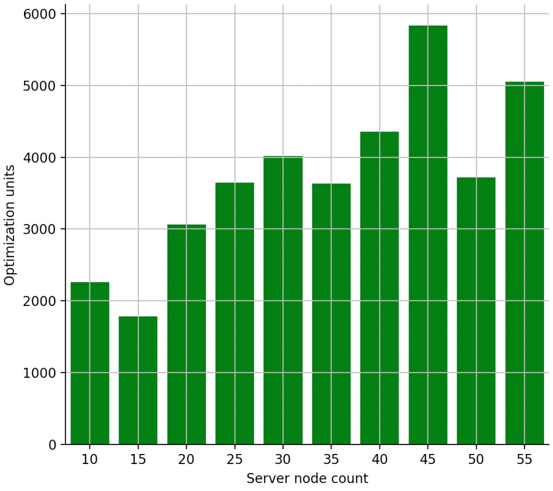


Figure 8: Optimization result difference between non-power optimized and power-optimized ILP implementation by server node count plot

On the other hand, the optimization unit difference in Figure-6 shows that the difference in power consumption between the two algorithms follows a generally increasing pattern as the server count increases. These units are calculated by our objective function of the ILP model. These results show that our proposed improvements provide **a high value of savings in power consumption**.

Furthermore, as can be seen in Figure-7, while there is no considerable amount of optimization time difference between non-optimized and optimized ILP algorithms, Figure-8 shows a huge value of power consumption difference between the two algorithms. Considering the greedy algorithm results from Figure-6, we can say that unless there is an optimization target, using the greedy approach rather than ILP could perform better. However, with the power optimization perspective, our ILP solution results in a huge amount of power savings.

6 Conclusion

In this paper, an ILP-based model was proposed to solve the power optimization problem in end-to-end network slice embedding under strict security requirements. Because of using binary variables in the model and simple objective functions, our model is suitable for large problem instances. The data generator we have built has enabled us to test the model with different network topologies of different sizes and parameter values. Through simulation experiments, we have demonstrated that the proposed model provides important power savings while meeting the strict QoS and security requirements.

As future work, some improvements to our model will be performed. Our current ILP model is suitable for static network slicing environments. However, real-time telecommunication networks are highly dynamic due to the constantly changing environment. For this purpose, dynamic slice creation or deletion will be considered in the optimization model. Deep reinforcement learning algorithms have proven to work successfully in dynamic environments for various problems. Hence, these algorithms will also be evaluated to add dynamicity to our modeling.

Acknowledgments

This study is partially supported by Vodafone within the framework of 5G and Beyond Joint Graduate Support Programme coordinated by Information and Communication Technologies Authority.

References

[1] M. Alizadeh, K. Andersson, and O. Schelen. A survey of secure internet of things in relation to blockchain. *Journal of Internet Services and Information Security (JISIS)*, 10(3):47–75, August 2020.

[2] I. Kholod, A. Shorov, and S. Gorlatch. Efficient distribution and processing of data for parallelizing data mining in mobile clouds. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 11(1):2–17, March 2020.

[3] H. Kim. 5g core network security issues and attack classification from network protocol perspective. *Journal of Internet Services and Information Security (JISIS)*, 10(2):1–15, May 2020.

[4] S. Nowaczewski and W. Mazurczyk. Securing future internet and 5g using customer edge switching using dnscrypt and dnssec. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 11(3):87–106, September 2020.

[5] X. Li, M. Samaka, H.A. Chan, D. Bhamare, L. Gupta, C. Guo, and R. Jain. Network slicing for 5g: Challenges and opportunities. *IEEE Internet Computing*, 21(5):20–27, September 2017.

[6] S. Buzzi, C. I, T.E. Klein, H.V. Poor, C. Yang, and A. Zappone. A survey of energy-efficient techniques for 5g networks and challenges ahead. *IEEE Journal on Selected Areas in Communications*, 34(4):697–709, April 2016.

[7] A.M. Khedr, P.R.P. V, and A.A. Ali. An energy-efficient data acquisition technique for hierarchical cluster-based wireless sensor networks. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, 11(3):70–86, September 2020.

[8] A. Fendt, S. Lohmuller, L.C. Schmelz, and B. Bauer. A network slice resource allocation and optimization model for end-to-end mobile networks. In *Proc. of the 2018 IEEE 5G World Forum (5GWF'18), Silicon Valley, CA, USA*, pages 262–267. IEEE, July 2018.

[9] H. Jmila and G. Blanc. Towards security-aware 5g slice embedding. *Computers & Security*, 100:102075, January 2021.

[10] R. Doriguzzi-Corin, S. Scott-Hayward, D. Siracusa, and E. Salvadori. Application-centric provisioning of virtual security network functions. In *Proc. of the 2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN'17), Berlin, Germany*, pages 276–279. IEEE, December 2017.

[11] J. Guan, Z. Wei, and I. You. Grbc-based network security functions placement scheme in sds for 5g security. *Journal of Network and Computer Applications*, 114:48–56, July 2018.

[12] L. D. Nguyen. Resource allocation for energy efficiency in 5g wireless networks. *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, 5(14):154832, June 2018.

[13] B. Matthiesen, O. Aydin, and E.A. Jorswieck. Throughput and energy-efficient network slicing. In *Proc. of the 22nd International ITG Workshop on Smart Antennas (WSA'18), Bochum, Germany*, pages 1–6. VDE, June 2018.

[14] Q. Shi, L. Zhao, Y. Zhang, G. Zheng, F. R. Yu, and H. Chen. Energy-efficiency versus delay tradeoff in wireless networks virtualization. *IEEE Transactions on Vehicular Technology*, 67(1):837–841, January 2018.

[15] Z. Liu, X. Chen, Y. Chen, and Z. Li. Deep reinforcement learning based dynamic resource allocation in 5g ultra-dense networks. In *Proc. of the 2019 IEEE International Conference on Smart Internet of Things (SmartIoT'19), Tianjin, China*, pages 168–174. IEEE, August 2019.

[16] F.E. Salem, Z. Altman, A. Gati, T. Chahed, and E. Altman. Reinforcement learning approach for advanced sleep modes management in 5g networks. In *Proc. of the 88th IEEE Vehicular Technology Conference (VTC-Fall'18), Chicago, IL, USA*, pages 1–5. IEEE, August 2018.

[17] M. Laroui, M.A. Cherif, H.I. Khedher, H. Moungla, and H. Afifi. Scalable and cost efficient resource allocation algorithms using deep reinforcement learning. In *Proc. of the 2020 International Wireless Communications and Mobile Computing (IWCMC'20), Limassol, Cyprus*, pages 946–951. IEEE, June 2020.

[18] R. Li, Z. Zhao, Q. Sun, I. Chih-Lin, C. Yang, X. Chen, M. Zhao, and H. Zhang. Deep reinforcement learning for resource management in network slicing. *IEEE Access*, 6:74429 – 74441, November 2018.

[19] C. Qi, Y. Hua, R. Li, Z. Zhao, and H. Zhang. Deep reinforcement learning with discrete normalized advantage functions for resource management in network slicing. *IEEE Communications Letters*, 23(8):1337–1341, June 2019.

[20] L. Zhao and L. Li. Reinforcement learning for resource mapping in 5g network slicing. In *Proc. of the 5th International Conference on Computer and Communication Systems (ICCCS'20), Shanghai, China*, pages 869–873. IEEE, June 2020.

[21] 3GPP. Technical Specification Group Services and System Aspects; System architecture for the 5G System (5GS); Stage 2. Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), March 2020. Version 16.4.0.

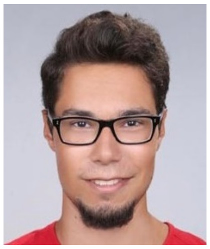
[22] A.A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines. 5g network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges. *Computer Networks*, 167:106984, February 2020.

[23] ETSI. European Telecommunications Standards Institute. GS NFV 002, European Telecommunications Standards Institute, 10 2013. Version 1.1.1.

Author Biography

Ozan Akin is a senior computer engineering student at Middle East Technical University, Ankara Turkey. He is interested in the topics networking, software defined networks, distributed-systems, cloud-computing, and language processors. He has been taking graduate courses to improve his knowledge in these subjects, such as Ad-Hoc Computing, Distributed Systems, and Language Processors at METU. He has been working as a backend developer and devops consultant for several companies for about three years. Also, he is currently in a 5G network slicing resource optimization research group.

Umut Can Gulmez is a computer engineering master degree student in Middle East Technical University. He also completed his B.S in Computer Engineering at METU in 2019. After graduation, Umut had worked for Vodafone Turkey over two years as a 5G Researcher in the 5G and Beyond Joint Graduate Support Program which was coordinated by the Information and Communication Technologies Authority Turkey. His research interests include 5G networks, Deep Reinforcement Learning and network slicing security. He is working as an Autonomous Vehicles Software Developer at Ford Otosan right now.



Ozan Sazak is a senior computer engineering student at Middle East Technical University, Turkey, and currently working as a Backend SWE. His research interests are distributed computing, cloud computing, operating systems and compilers. He has been taking graduate courses on distributed systems, language processors, and advanced operating systems to broaden his knowledge in these fields. He is currently in a 5G network slicing and resource optimization research group at METU, and the focus of his current work is cloud computing and software-defined networks.

Ufuk Yagmur is a senior computer engineering student at Middle East Technical University, Ankara Turkey. He has taken graduate courses such as Ad Hoc Computing, Wireless Networks and Distributed Computing. In addition, he is currently working on a 5G Network Slicing optimization research project. In this project, a conference paper was published and currently working on the extension of that paper. He had an internship in full-stack Web-App development and Cyber-Security.

Pelin Angin is an Assistant Professor of Computer Engineering at Middle East Technical University. She completed her B.S. in Computer Engineering at Bilkent University in 2007 and her Ph.D. in Computer Science at Purdue University, USA in 2013. Between 2014-2016, she worked as a Visiting Assistant Professor and Postdoctoral Researcher at Purdue University. Her research interests lie in the fields of cloud computing and IoT security, distributed systems, 5G networks and blockchain. She is among the founding members of the Systems Security Research Laboratory and an affiliate of the Wireless Systems, Networks and Cybersecurity Laboratory at METU. She serves on the editorial boards of multiple journals on IoT and mobile computing. Her work in security has been published at high impact journals including IEEE Transactions on Dependable and Secure Computing, Computers & Security and IEEE Access among others.